

App. No. 09/609,494
Amendment Dated: April 29, 2004
Reply to Office Action of February 9, 2004

Amendments to the Claims:

Claim 1: (original) A method for enhancing flow analysis, comprising:
inferring types from a program;
forming a type graph having polarities and indices from the types; and
forming a flow graph from the type graph to inhibit imprecise paths so as to enhance context-sensitivity of flow analysis.

Claim 2: (original) The method of claim 1, wherein inferring includes generating constraints from the program and solving the constraints.

Claim 3: (original) The method of claim 2, wherein generating includes generating the constraints, wherein the constraints include a set of equalities and inequalities, wherein the set is a finite set, and wherein the set of equalities and inequalities is adapted to be a set of simultaneous equations.

Claim 4: (original) The method of claim 3, wherein generating includes generating the constraints, wherein an equality from the set of equalities defines that a type is equal to another type, and wherein the type and the another type are adapted to be unified.

Claim 5: (original) The method of claim 3, wherein generating includes generating the constraints, wherein an inequality from the set of inequalities defines that a type is an instance of another type, wherein the inequality includes an index and a polarity.

Claim 6: (original) A computer readable medium having instructions stored thereon for causing a computer to perform a method for enhancing flow analysis, the method comprising:
inferring types from a program;
forming a type graph having polarities and indices from the types; and
forming a flow graph from the type graph so as to enhance context-sensitivity of flow analysis.

App. No. 09/609,494
Amendment Dated: April 29, 2004
Reply to Office Action of February 9, 2004

Claim 7: (original) A method for enhancing flow analysis, comprising:
forming a type graph that includes polarities and indices; and
forming a flow graph that includes a set of flow paths, wherein the set of flow paths excludes imprecise paths so as to enhance context-sensitivity of flow analysis.

Claim 8: (original) The method of claim 7, wherein forming the set of flow paths includes forming at least one flow path that inherits a polarity from the polarities of the type graph.

Claim 9: (original) The method of claim 8, wherein forming includes forming the set of flow paths, wherein each imprecise path includes two flow edges, wherein the polarity of one of the two flow edges is negative, and wherein the polarity of the other of the two flow edges is positive.

Claim 10: (original) The method of claim 9, wherein forming includes forming the set of flow paths, wherein each imprecise path is defined by the one of the two flow edges preceding the other of the two flow edges.

Claim 11: (original) The method of claim 7, wherein forming includes forming the type graph, wherein each polarity indicates at least one of a value flowing into a type and a value flowing out of a type.

Claim 12: (original) A computer readable medium having instructions stored thereon for causing a computer to perform a method for enhancing flow analysis, the method comprising:
forming a type graph that includes polarities and indices; and
forming a flow graph that includes a set of flow paths, wherein the set of flow paths excludes imprecise paths so as to enhance context-sensitivity of flow analysis.

Claim 13: (original) A method for forming a graph to enhance flow analysis, comprising:
forming a plurality of nodes to represent types; and

App. No. 09/609,494
Amendment Dated: April 29, 2004
Reply to Office Action of February 9, 2004

forming a plurality of instantiation paths to represent instantiations of generic types to instance types, wherein each instantiation path includes an index and a polarity.

Claim 14: (original) The method of claim 13, further comprising forming a plurality of flow paths to represent a flow of values between types.

Claim 15: (original) The method of claim 14, wherein forming includes forming a plurality of flow paths, wherein each flow path includes a polarity that is inherited from an instantiation path.

Claim 16: (original) The method of claim 15, wherein forming includes forming a plurality of flow paths, wherein the polarity defines at least one of an ingress and an egress flow of a value with respect to a type.

Claim 17: (original) The method of claim 13, wherein forming includes forming a plurality of instantiation paths, wherein the index defines an occurrence of an instantiation of a generic type to an instance type so as to differentiate among occurrences of instantiations of the generic type to the instance type.

Claim 18: (original) A computer readable medium having instructions stored thereon for causing a computer to perform a method for forming a graph to enhance flow analysis, the method comprising:

forming a plurality of nodes to represent types; and

forming a plurality of instantiation paths to represent instantiations of generic types to instance types, wherein each instantiation path includes an index and a polarity.

Claims 19 – 24 (Canceled).

Claim 25: (original) A method for inferring types to enhance flow analysis, comprising: generating constraints from a program; and

App. No. 09/609,494
Amendment Dated: April 29, 2004
Reply to Office Action of February 9, 2004

solving the constraints to infer at least one type, wherein solving includes propagating polarities so as to enhance flow analysis.

Claim 26: (original) The method of claim 25, wherein propagating includes propagating polarities, wherein each polarity is adapted to indicate at least one of a negative polarity, a positive polarity, and a bidirectional polarity.

Claim 27: (original) The method of claim 25, wherein propagating includes propagating polarities, wherein the negative polarity defines a value flowing into a type.

Claim 28: (original) The method of claim 25, wherein propagating includes propagating polarities, wherein the positive polarity defines a value flowing out of a type.

Claim 29: (original) The method of claim 25, wherein propagating includes propagating polarities, wherein the bi-directional polarity defines a combination of a negative polarity and a positive polarity.

Claim 30: (original) A computer readable medium having instructions stored thereon for causing a computer to perform a method for inferring types to enhance flow analysis, the method comprising:

generating constraints from a program; and

solving the constraints to infer at least one type, wherein solving includes propagating polarities so as to enhance flow analysis.

Claim 31: (original) A method for enhancing flow analysis, comprising:
annotating each expression in a program by a label;
associating the label of an expression with a type of the expression; and
tracing at least one path on a type graph having polarities to determine if a value arising at one label in the program flows to another label in the program.

App. N . 09/609,494
Amendment Dated: April 29, 2004
Reply to Office Action of February 9, 2004

Claim 32: (original) A computer readable medium having instructions stored thereon for causing a computer to perform a method for enhancing flow analysis, the method comprising:
annotating each expression in a program by a label;
associating the label of an expression with a type of the expression; and
tracing at least one path on a type graph having polarities to determine if a value arising at one label in the program flows to another label in the program.

Claim 33: (original) A method for enhancing flow analysis, comprising:
forming a type instantiation graph that includes polarities and indices; and
computing points-to information for at least one program point by answering reachability queries on the type instantiation graph.

Claim 34: (original) The method of claim 33, wherein forming includes forming with a polymorphic inference technique based on instantiation constraints.

Claim 35: (original) The method of claim 33, wherein answering includes answering reachability queries within a desired duration, wherein the desired duration is linearly proportional to a size of the type instantiation graph.

Claim 36: (original) A computer readable medium having instructions stored thereon for causing a computer to perform a method for enhancing flow analysis, the method comprising:
forming a type instantiation graph that includes polarities and indices; and
computing points-to information for at least one program point by answering reachability queries on the type instantiation graph.

Claim 37 - 41 (Canceled)

Claim 42: (original) A method for enhancing flow analysis, comprising:
forming a type instantiation graph that includes polarities and indices;
forming a flow graph to form a set of flow paths; and

App. No. 09/609,494
Amendment Dated: April 29, 2004
Reply to Office Action of February 9, 2004

forming a subset from the set of flow paths such that context-sensitivity of flow analysis is enhanced.

Claim 43: (original) The method of claim 42, wherein forming includes forming a set of flow paths that includes at least one imprecise path, wherein the subset excludes the at least one imprecise path.

Claim 44: (original) The method of claim 43, wherein forming the type-instantiation graph includes forming a plurality of nodes, wherein each node of the plurality of nodes represents a type expression.

Claim 45: (original) The method of claim 44, wherein forming includes forming a plurality of instantiation paths, wherein each instantiation path represents an instantiation of a generic type expression to an instance type expression, wherein each instantiation path includes an index and a polarity, wherein the index represents an occurrence of an instantiation of the generic type expression to the instance type expression.

Claim 46: (original) The method of claim 45, wherein forming includes forming a set of flow paths such that each flow path includes a polarity, wherein the polarity of the flow path is positive if the polarity of the instantiation path is positive, and wherein the polarity of the flow path is negative if the polarity of the instantiation path is negative.

Claim 47: (original) The method of claim 46, wherein forming a set of flow paths includes forming a path, wherein the path includes at least two flow edges.

Claim 48: (original) The method of claim 47, wherein forming a path includes forming a path, wherein the path is defined to be an imprecise path if a polarity of one of the at least two flow edges is negative and a polarity of an other of the at least two flow edges is positive, and wherein the one of the at least two flow edges precedes the other of the at least two flow edges.

App. No. 09/609,494
Amendment Dated: April 29, 2004
Reply to Office Action of February 9, 2004

Claim 49: (original) A computer readable medium having instructions stored thereon for causing a computer to perform a method for enhancing flow analysis, the method comprising:
forming a type instantiation graph that includes polarities and indices;
forming a flow graph to form a set of flow paths; and
forming a subset from the set of flow paths such that context-sensitivity of flow analysis is enhanced.

Claim 50 - 56: (Canceled)

Claim 57: (New) A method for enhancing flow analysis, comprising: abstracting program expressions into types; and forming a type graph from the types, wherein the type graph includes polarities and indices so as to enhance flow analysis, wherein abstracting includes unifying recursive types, wherein unifying is adapted to be executed using a cyclic unification technique.

Claim 58: (New) A method for enhancing flow analysis, comprising: abstracting program expressions into types; and forming a type graph from the types, wherein the type graph includes polarities and indices so as to enhance flow analysis, wherein forming includes forming a plurality of nodes, wherein each node represents a type, and wherein forming includes forming at least one instantiation path between two nodes to represent an instantiation of a generic type to an instance type.

Claim 59: (New) The method of claim 58, wherein forming includes forming at least one instantiation path that includes an index and a polarity.